NAME: NIMISH KUSHWAHA

REG. NO.: CH.EN.U4CSE22073

IMPLEMENTATION OF A QUEUE USING ARRAYS IN C LANGUAGE

1. PROGRAM CODE:

```c
#include<stdio.h>
#define MAX_SIZE 100

int queue[MAX_SIZE];
int front = -1;
int rear = -1;

int isEmpty(){
    return (front == -1 && rear == -1);
}
int isFull(){
    return (rear == MAX_SIZE -1);
}
void enqueue(int value){
    if (isFull()){
        printf("Queue is Full. Enqueue opeation is failed.\n");
        return;
    }

   else if(isEmpty()){
    front = 0;
    rear = 0;
   }
    else{
    rear++;
    }

   queue[rear] = value;
   printf("%d enqueued sucessfully.\n", value);
}
```

```c
// Function to dequeue an element from the front of the queue
int dequeue(){
    int value;
    if (isEmpty()){
        printf("Queue is empty. Dequeue operation failed.\n");
        return -1;

    } else {
        value = queue[front];
        if(front == rear){
            front = -1;
            rear = -1;
        } else {
            front++;
        }
        return value;
    }
}
//function to display the elements in the queue
void display(){
    if(isEmpty()){
        printf("Queue is empty.\n");
    }else{
        printf("Queue elements: ");
        for(int i = front; i<=rear; i++){
            printf("%d ", queue[i]);
        }
        printf("\n");
    }
}
int main(){
    enqueue(10);
    enqueue(20);
    enqueue(30);
    display(); //Queue elements: 10 20 30

    int dequeudValue = dequeue();
    printf("Dequeued value: %d\n", dequeudValue); //Dequeued value: 10
    return 0;
}
```

2. OUTPUT:

```
PS D:\My Code\FDS> gcc q.c
PS D:\My Code\FDS> ./a.exe
10 enqueued sucessfully.
20 enqueued sucessfully.
30 enqueued sucessfully.
Queue elements: 10 20 30
Dequeued value: 10
PS D:\My Code\FDS>
```

IMPLEMENTATION OF A QUEUE USING LINKED LIST IN C LANGUAGE

1. PROGRAM CODE:

```c
#include<stdio.h>
#include<stdlib.h>

//Node structure for the linked list
struct Node{
    int data;
    struct Node* next;
};

struct Node* front = NULL; //Front pointer of the queue
struct Node* rear = NULL; //Rear pointer of the queue

//Function to check if the queue is empty
int isEmpty(){
    return (front == NULL);
}

//Function to enqueue an element to the rear of the queue
void enqueue(int value){
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = value;
    newNode->next = NULL;

    if (rear == NULL){
        front = newNode;
        rear = newNode;
    } else{
        rear->next = newNode;
        rear = newNode;
    }
    printf("\n%d enqueued successfully.", value);
}
```

```c
//Function to dequeue an element from the front of the queue
int dequeue(){
    if(isEmpty()){
        printf("Queue is empty. Dequeue operation failed.\n");
        return -1;
    } else{
        struct Node* temp = front;
        int value = front->data;
        front = front->next;
        free(temp);

        if(front == NULL){
            rear = NULL;
        }
        return value;
    }
}

//Function to display the elements in the queue
void display(){
    if (isEmpty()){
        printf("Queue is empty.\n");
    } else {
        printf("\nQueue elements: ");
        struct Node* temp = front;
        while (temp != NULL){
            printf("%d ", temp->data);
            temp = temp->next;
        }
        printf("\n");
    }
}
```

```c
int main(){
    enqueue(10);
    enqueue(20);
    enqueue(30);
    display(); //Queue elements: 10 20 30

    int dequeuedValue = dequeue();
    printf("Dequeued value: %d\n", dequeuedValue); //Dequeued value: 10
    printf("\n");

    return 0;
}
```

2. OUTPUT:

```
PS D:\My Code\FDS> gcc queue.c
PS D:\My Code\FDS> ./a.exe

10 enqueued successfully.
20 enqueued successfully.
30 enqueued successfully.
Queue elements: 10 20 30
Dequeued value: 10
```

IMPLEMENTATION OF A STACK USING AN ARRAY IN C LANGUAGE:

1. PROGRAM CODE:

```c
#include<stdio.h>
int stack[100],choice,n,top,x,i;
void push(void);
void pop(void);
void display(void);

void push()
{
    if(top>=n-1)
    {
        printf("\nSTACK is over flow");

    }
    else
    {
        printf(" Enter a value to be pushed:");
        scanf("%d",&x);
        top++;
        stack[top]=x;
    }
}

void pop()
{
    if(top<=-1)
    {
        printf("\n Stack is under flow");
    }
    else
    {
        printf("\n The popped elements is %d",stack[top]);
        top--;
    }
}
```

```c
void display()
{
    if(top>=0)
    {
        printf("\n The elements in STACK \n");
        for(i=top; i>=0; i--)
            printf("\n%d",stack[i]);
        printf("\n Press Next Choice");
    }
    else
    {
        printf("\n The STACK is empty");
    }

}

int main()
{
    top=-1;
    printf("\n Enter the size of STACK[MAX=100]:");
    scanf("%d",&n);
    printf("\nSTACK OPERATIONS USING ARRAY");
    printf("\n 1.PUSH\n 2.POP\n 3.DISPLAY\n 4.EXIT");
    do
    {
        printf("\nEnter the Choice:");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:
            {
                push();
                break;
            }
            case 2:
            {
                pop();
                break;
            }
            case 3:
            {
                display();
                break;
            }
            case 4:
            {
                printf("\nEXIT POINT ");
                break;
            }
            default:
            {
                printf ("\n Please Enter a Valid Choice(1/2/3/4)");
            }

        }
    }
    while(choice!=4);
    return 0;
}
```

2. OUTPUT:

```
Enter the size of STACK[MAX=100]:4

STACK OPERATIONS USING ARRAY
 1.PUSH
 2.POP
 3.DISPLAY
 4.EXIT
Enter the Choice:1
 Enter a value to be pushed:5

Enter the Choice:1
 Enter a value to be pushed:4

Enter the Choice:1
 Enter a value to be pushed:3

Enter the Choice:1
 Enter a value to be pushed:2

Enter the Choice:3

 The elements in STACK

2
3
4
5
```

```
 Press Next Choice
Enter the Choice:2

 The popped elements is 2
Enter the Choice:3

 The elements in STACK

3
4
5
 Press Next Choice
Enter the Choice:2

 The popped elements is 3
Enter the Choice:3

 The elements in STACK

4
5
 Press Next Choice
Enter the Choice:4

EXIT POINT
PS D:\Study Materials\2nd Sem\19CSE111 (FDS)\Exercise in C>
```